What Is Claimed Is:

1. A method of caching a data object, comprising:

receiving at a first cache of a plurality of cooperating caches a first data object of a domain of data objects;

if said first data object is owned by the first cache, storing said first data object in the first cache; and

if said first data object is owned by another cache in the plurality of caches, determining on the basis of a set of dynamic criteria whether to store said first data object in the first cache;

wherein said first data object is owned by one and only one of the plurality of caches.

2. The method of claim 1, further comprising:

identifying one of the plurality of caches as the owner of said first data object.

3. The method of claim 2, wherein said identifying comprises:

hashing an identifier of said first data object to produce a hash value; and

mapping said hash value to one of said plurality of caches.

4. The method of claim 1, wherein said receiving comprises receiving said first data object from said other cache in the plurality of caches.

5. The method of claim 1, wherein said set of dynamic criteria includes a popularity of said first data object.

6. The method of claim 1, wherein said set of dynamic criteria includes a utilization of the first cache.

7. The method of claim 1, wherein said set of dynamic criteria includes a size of said first data object.

8. The method of claim 1, further comprising:

removing a cached data object from the first cache;

wherein said cached data object is selected based on one or more criteria.

9. The method of claim 8, wherein said one or more characteristics include popularity;

wherein said popularity is measured as one or more of:

a number of requests for said cached data object; and

a frequency of requests for said cached data object.

10. The method of claim 8, wherein said one or more criteria include validity.

11. The method of claim 8, wherein said one or more criteria include age.

12. The method of claim 8, wherein said one or more criteria include size.

13. The method of claim 8, wherein said one or more criteria include ownership.

14. The method of claim 8, wherein said one or more criteria include a cost of retrieving said cached data object from one of an origin server and a second cache in the plurality of caches.

15. The method of claim 8, wherein said one or more criteria include a level of storage input/output activity at the first cache.

16. The method of claim 8, wherein said one or more criteria include a level of communication activity at the first cache.

17. The method of claim 8, wherein said one or more criteria include a level of processor activity at the first cache.

18. The method of claim 1, further comprising:

propagating invalidation of said first data object between the first cache and a second cache.

19. The method of claim 1, further comprising:

exchanging a configuration of the plurality of cooperating caches between the first cache and a second cache.

20. The method of claim 1, further comprising:

re-configuring ownership of the domain of data objects in response to the removal of a cache from the plurality of cooperating caches.

21. The method of claim 1, further comprising:

re-configuring ownership of the domain of data objects in response to the addition of a cache to the plurality of cooperating caches.

22. A computer readable storage medium storing instructions that, when executed by a computer, cause the computer to perform a method of caching a data object, the method comprising:

receiving at a first cache of a plurality of cooperating caches a first data object of a domain of data objects;

if said first data object is owned by the first cache, storing said first data object in the first cache; and

if said first data object is owned by another cache in the plurality of caches, determining on the basis of a set of dynamic criteria whether to store said first data object in the first cache;

wherein said first data object is owned by one and only one of the plurality of caches.

23. A method of caching data objects in a plurality of cooperating caches, comprising:

partitioning a set of data objects among a plurality of cooperating caches, wherein each of said caches receives ownership of a subset of said data objects;

caching one or more data objects of a first subset of said data objects at a first cache having ownership of said first subset;

caching one or more data objects of a second subset of said data objects at the first cache, wherein a second cache in the cluster owns said second subset;

receiving at the first cache a first request for a first data object in said second subset of data objects;

receiving said first data object from the second cache; and

caching said first data object at the first cache only if said first data object satisfies one or more of a predetermined set of criteria.

24. The method of claim 23, wherein said caching said first data object comprises caching said first data object if said first data object has a threshold level of popularity.

25. The method of claim 23, wherein said caching said first data object comprises caching said first data object if the first cache has capacity to cache said first data object without first removing another data object.

26. The method of claim 23, further comprising:

removing one or more cached data objects from the first cache, wherein a subset of said set of criteria is used to select said one or more cached data objects.

27. The method of claim 23, wherein said predetermined set of criteria includes a popularity of said first data object.

28. The method of claim 23, wherein said predetermined set of criteria includes a validity of said first data object.

29. The method of claim 23, wherein said predetermined set of criteria includes a size of said first data object.

30. The method of claim 23, wherein said predetermined set of criteria includes an age of said first data object.

31. The method of claim 23, wherein said predetermined set of criteria includes a cost of retrieving said first data object from an origin server.

32. The method of claim 23, wherein said predetermined set of criteria includes a measure of the utilization of the first cache.

33. The method of claim 23, further comprising:

receiving an invalidation message regarding said first data object at one of the first cache and the second cache; and

communicating said invalidation to the other of the second cache and the first cache.

34. The method of claim 23, further comprising:

automatically re-partitioning ownership of the set of data objects upon failure of one of the cooperating caches.

35. The method of claim 23, further comprising:

automatically re-partitioning ownership of the set of data objects upon the addition of a cache to the plurality of cooperating caches.

36. A computer readable storage medium storing instructions that, when executed by a computer, cause the computer to perform a method of caching data objects in a plurality of cooperating caches, the method comprising:

partitioning a set of data objects among a plurality of cooperating caches, wherein each of said caches receives ownership of a subset of said data objects;

caching one or more data objects of a first subset of said data objects at a first cache having ownership of said first subset;

caching one or more data objects of a second subset of said data objects at the first cache, wherein a second cache in the cluster owns said second subset;

receiving at the first cache a first request for a first data object in said second subset of data objects;

receiving said first data object from the second cache; and

caching said first data object at the first cache only if said first data object satisfies one or more of a predetermined set of criteria.

37. A method of caching data objects in a plurality of cooperating caches, comprising:

partitioning a domain of data objects among a plurality of cooperating caches, wherein a first cache receives ownership of a first subset of said data objects;

caching one or more members of said first subset of data objects at the first cache;

caching one or more members of a second subset of data objects at the first cache, wherein a second cache owns said second subset of data objects; and

removing a first cached data object from said first cache, wherein said first data object is identified by applying a predetermined set of criteria.

38. The method of claim 37, wherein said predetermined set of criteria includes data object popularity.

39. The method of claim 37, wherein said predetermined set of criteria includes data object validity.

40. The method of claim 37, wherein said predetermined set of criteria

includes data object size.

41. The method of claim 37, wherein said predetermined set of criteria includes data object age.

42. The method of claim 37, wherein said predetermined set of criteria includes data object ownership.

43. The method of claim 37, wherein said predetermined set of criteria includes a cost of retrieving a data object from an origin server.

44. The method of claim 37, wherein said predetermined set of criteria includes a measure of the utilization of the first cache.

45. The method of claim 37, further comprising:

receiving at the first cache an invalidation message regarding a data object cached in the first cache; and

communicating said invalidation of said data object to another cache.

46. A computer readable storage medium storing instructions that, when executed by a computer, cause the computer to perform a method of caching data objects in a plurality of cooperating caches, the method comprising:

partitioning a domain of data objects among a plurality of cooperating caches, wherein a first cache receives ownership of a first subset of said data objects;

caching one or more members of said first subset of data objects at the first cache;

caching one or more members of a second subset of data objects at the first cache, wherein a second cache owns said second subset of data objects; and

removing a first cached data object from said first cache, wherein said first data object is identified by applying a predetermined set of criteria.

47. A hybrid cache, comprising:

a cache engine configured to cache a first subset of a domain of data objects, wherein ownership of said first subset of data objects is assigned to the hybrid cache;

a monitor configured to monitor an operational status of the hybrid cache;

an administrator configured to facilitate administration of the hybrid cache; and

communication links coupling the hybrid cache to one or more other hybrid caches;

wherein said cache engine is further configured to cache a second data object owned by a second hybrid cache if said second data object satisfies a set of dynamic criteria.

48. The hybrid cache of claim 47, wherein said domain of data objects is partitioned among the hybrid cache and the other hybrid caches such that each said cacheable data object is owned by just one of the hybrid caches.

49. The hybrid cache of claim 47, wherein said dynamic criteria include one or more of: popularity, validity, age, size, ownership and cost of retrieving said second data object.

50. The hybrid cache of claim 47, wherein one or more of said cache

engine and said monitor are configured to report the invalidation of said second data object to the second hybrid cache.

51. A cluster of hybrid caches, comprising:

a plurality of hybrid caches;

a set of data objects, wherein ownership of said data objects is partitioned among said hybrid caches; and

a set of criteria for applying to determine whether to cache at a first hybrid cache a data object owned by a second hybrid cache;

wherein each of said hybrid caches is configured to always cache a first received data object that it owns and to apply said set of criteria to determine whether to cache a second received data object that belongs to a different hybrid cache.